

**Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)**

Колледж информатики и программирования


ОТЧЁТ
По учебной практике

Специальность 09.02.07 «Информационные системы и программирование»
(код) (наименование)

Профессиональный модуль ПМ.01 Разработка модулей программного обеспечения для компьютерных систем
(код) (наименование)

Междисциплинарный курс МДК.01.01 Разработка программных модулей
(код) (наименование)

Выполнил:

Студент 4 курса 4ИСИП-421 учебной группы
(номер) (номер)
 Д.А. Мамонтов
(подпись) (инициалы, фамилия)

Проверил:

Руководитель практики от Колледжа информатики и программирования
Преподаватель 1КК, к.т.н., доцент Е.Л. Альшакова
(квалификационная категория или звание, должность) (инициалы, фамилия)

Москва – 2024

Перечень работ, выполненных в ходе учебной практики

№ п/п	Виды работ	Оценка
1	Проектирование базы данных и разработка объектов базы данных	Отлично
2	Разработка приложения с использованием WPF	Отлично

СОДЕРЖАНИЕ

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1.....	4
ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2.....	14

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 1

Тема: Проектирование базы данных и разработка объектов базы данных

Студент: Д.А. Мамонтов

Группа: 4ИСИП-421

Преподаватель: Альшакова Е.Л.

Дата: 18.12.2024

ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ И РАЗРАБОТКА ОБЪЕКТОВ БАЗЫ

Цель работы: разработать базу данных в соответствии с заданием, наполнить ее сущностями и атрибутами, сделать ER-диаграмму.

На рисунке 1 представлена созданная ER-диаграмма БД EquipmentRepair.

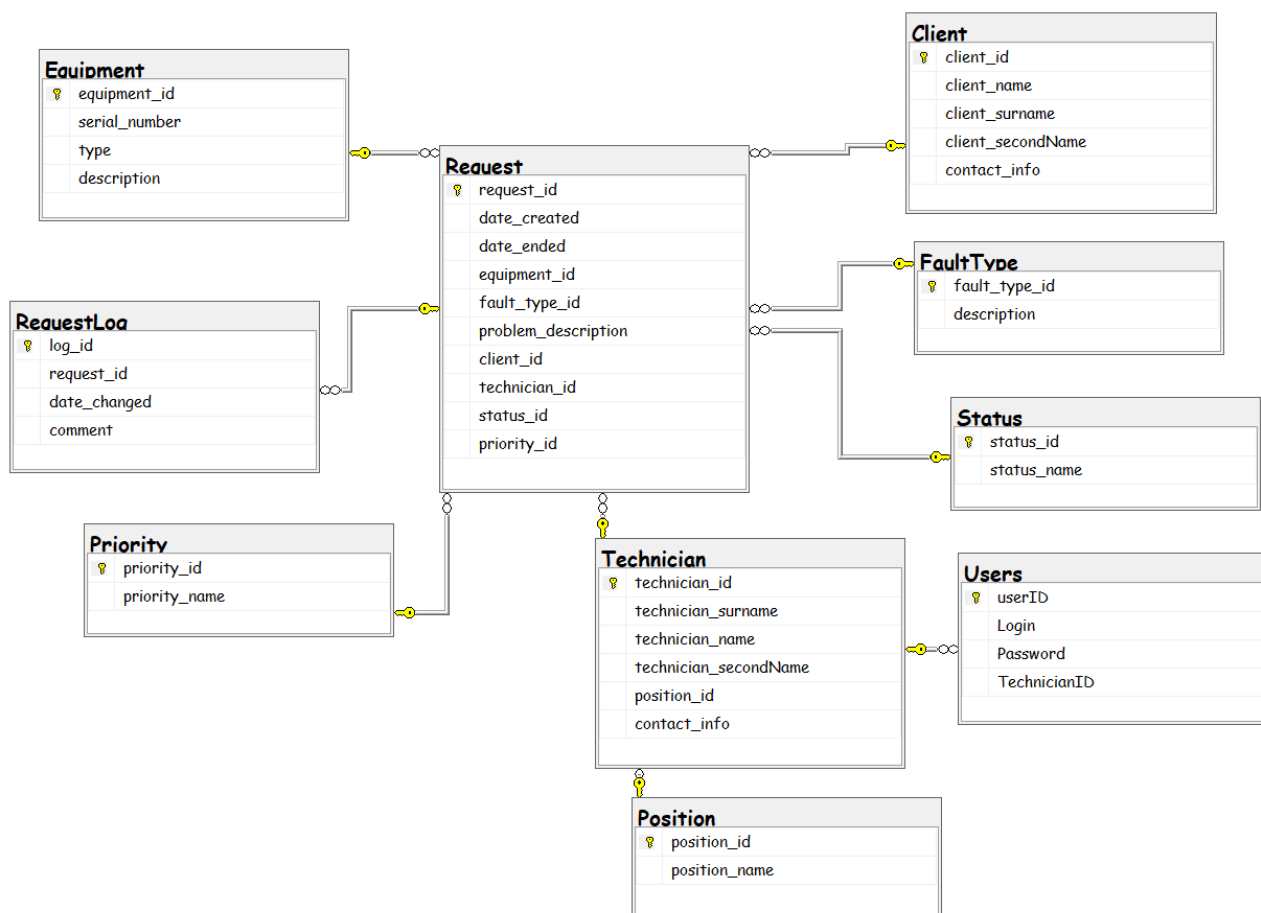


Рисунок 1 – ER диаграмма

На рисунках 2-11 представлено наполнение таблиц БД.

Results		Messages			
	client_id	client_name	client_surname	client_secondName	contact_info
1	1	Иван	Иванов	Иванович	+79123456789
2	2	Мария	Петрова	Петровна	+79876543210
3	3	Александр	Сидоров	Александрович	+79012345678
4	4	Ольга	Козлова	Владимировна	+79999999999
5	5	Андрей	Кузнецов	Сергеевич	+79000000000
6	6	Иван	Иванов	Иванович	+79123456789
7	7	Мария	Петрова	Петровна	+79876543210

Рисунок 2 – Таблица Client

Results Messages				
	equipment_id	serial_number	type	description
1	1	EQ12345	Станок	Токарный станок
2	2	EQ12346	Принтер	Цветной принтер
3	3	EQ12347	Сервер	Сервер DELL
4	4	EQ12348	Сканер	Сканер Phillips
5	5	EQ12349	Паяльник	Паяльник
6	6	EQ12350	Сканер	Сканер

Рисунок 3 – Таблица Equipment

Results Messages						
	technician_id	technician_surname	technician_name	technician_secondName	position_id	contact_info
1	1	Иванов	Иван	Петрович	1	smirnov@yandex.ru
2	2	Петров	Петр	Иванович	2	kuznetsov@yandex.ru
3	3	Сидоров	Сидор	Павлович	3	novikov@yandex.ru
4	4	Кузнецов	Дмитрий	Александрович	1	nikitin@yandex.ru
5	5	Смирнов	Алексей	Сергеевич	3	loshkin@yandex.ru

Рисунок 4 – Таблица Technician

Results Messages		
	fault_type_id	description
1	1	Не включается
2	2	Выдает ошибку
3	3	Не печатает
4	4	Сбой в сети
5	5	Синий экран смерти
6	6	Нет подключения к интернету
7	7	Медленная работа

Рисунок 5 – Таблица FaultType

Results Messages										
	request_id	date_created	date_ended	equipment_id	fault_type_id	problem_description	client_id	technician_id	status_id	priority_id
1	1	2024-01-12 10:00:00.000	2024-12-18 16:02:15.023	1	1	Ноутбук не включается	1	1	3	3
2	2	2024-02-12 12:00:00.000	2024-12-18 00:00:00.000	2	2	Смартфон перегревается при использовании камеры	2	2	3	2
3	3	2024-03-12 15:00:00.000	2024-12-20 00:00:00.000	3	5	Принтер не печатает	3	4	2	1
4	4	2024-04-12 09:00:00.000	2024-12-01 00:00:00.000	4	2	Не работает сканер	4	5	3	3
5	5	2024-05-12 14:00:00.000	2024-12-08 00:00:00.000	5	1	Не работает паяльник	5	2	3	2
6	6	2024-06-12 16:00:00.000	2024-12-15 00:00:00.000	2	3	Не печатает цветные фотографии	2	3	2	2
7	7	2024-07-12 11:00:00.000	2024-12-18 00:00:00.000	3	1	Нет изображения на планшете	4	2	1	2
8	8	2024-12-18 14:58:21.230	2024-12-12 00:00:00.000	2	2	Не работает	4	2	3	3
9	9	2024-12-18 00:00:00.000	2024-12-18 00:00:00.000	2	2	не работает	3	1	1	3
10	10	2024-12-04 00:00:00.000	2024-12-18 16:02:21.933	2	2	Не работает	2	1	3	1
11	11	2024-12-11 00:00:00.000	2024-12-18 00:00:00.000	6	5	Не работает	3	4	3	3
12	12	2024-12-11 00:00:00.000	2024-12-18 00:00:00.000	2	3	123	1	1	2	2

Рисунок 6 – Таблица Request

Results Messages		
	status_id	status_name
1	1	В ожидании
2	2	В работе
3	3	Завершено
4	4	Отменено

Рисунок 7 – Таблица Status

Results Messages		
	position_id	position_name
1	1	Менеджер
2	2	Инженер по ремонту
3	3	Системный администратор
4	4	Техник по обслуживанию
5	5	Инженер по безопасности
6	6	Программист
7	7	Руководитель отдела

Рисунок 8 – Таблица Position

Results Messages		
	priority_id	priority_name
1	1	Низкий
2	2	Средний
3	3	Высокий
4	4	Критический

Рисунок 9 – Таблица Priority

Results Messages				
	log_id	request_id	date_changed	comment
1	1	1	2024-01-12 11:00:00.000	Диагностика выполнена, требуется замена блока пит...
2	8	2	2024-02-12 13:30:00.000	Обновление ПО решило проблему перегрева
3	9	3	2024-12-03 20:57:40.123	Принтер настроен, заменены картриджи
4	10	4	2024-12-18 14:31:33.510	Клиенту отправлена инструкция по настройке роутера
5	11	5	2024-12-18 14:40:32.877	Монитор протестирован, проблема устранена
6	12	6	2024-12-18 14:40:47.740	Планшет восстановлен после сброса настроек
7	13	7	2024-12-18 14:40:54.447	Проблема с ПК устранена после замены оперативной...
8	14	8	2024-12-18 14:41:05.913	NULL
9	15	3	2024-12-18 14:47:59.063	NULL
10	16	1	2024-12-18 14:50:13.437	NULL

Рисунок 10 – Таблица RequestLog

Results Messages				
	userID	Login	Password	TechnicianID
1	1	ivanov	password123	1
2	2	petrov	qwerty456	2
3	3	sidorov	pass789	3
4	4	kuznetsov	admin123	1
5	5	smirnov	secure456	2

Рисунок 11 – Таблица Users

Скрипт для БД представлен в листинге 1.

Листинг 1 – Скрипт БД

```
USE [master]
GO
/***** Object:  Database [EquipmentRepair]      Script Date: 19.12.2024
16:06:09 *****/
CREATE DATABASE [EquipmentRepair]
    CONTAINMENT = NONE
    ON PRIMARY
    ( NAME = N'EquipmentRepair', FILENAME =
N'E:\Games\MSSQL15.MSSQLSERVER\MSSQL\DATA\EquipmentRepair.mdf' , SIZE = 8192KB
, MAXSIZE = UNLIMITED, FILEGROWTH = 65536KB )
    LOG ON
    ( NAME = N'EquipmentRepair_log', FILENAME =
N'E:\Games\MSSQL15.MSSQLSERVER\MSSQL\DATA\EquipmentRepair_log.ldf' , SIZE =
8192KB , MAXSIZE = 2048GB , FILEGROWTH = 65536KB )
    WITH CATALOG_COLLATION = DATABASE_DEFAULT
GO
ALTER DATABASE [EquipmentRepair] SET COMPATIBILITY_LEVEL = 120
GO
IF (1 = FULLTEXTSERVICEPROPERTY('IsFullTextInstalled'))
begin
EXEC [EquipmentRepair].[dbo].[sp_fulltext_database] @action = 'enable'
end
GO
ALTER DATABASE [EquipmentRepair] SET ANSI_NULL_DEFAULT OFF
GO
ALTER DATABASE [EquipmentRepair] SET ANSI_NULLS OFF
GO
ALTER DATABASE [EquipmentRepair] SET ANSI_PADDING OFF
GO
ALTER DATABASE [EquipmentRepair] SET ANSI_WARNINGS OFF
GO
ALTER DATABASE [EquipmentRepair] SET ARITHABORT OFF
GO
ALTER DATABASE [EquipmentRepair] SET AUTO_CLOSE OFF
GO
ALTER DATABASE [EquipmentRepair] SET AUTO_SHRINK OFF
GO
ALTER DATABASE [EquipmentRepair] SET AUTO_UPDATE_STATISTICS ON
GO
ALTER DATABASE [EquipmentRepair] SET CURSOR_CLOSE_ON_COMMIT OFF
GO
ALTER DATABASE [EquipmentRepair] SET CURSOR_DEFAULT GLOBAL
GO
ALTER DATABASE [EquipmentRepair] SET CONCAT_NULL_YIELDS_NULL OFF
GO
ALTER DATABASE [EquipmentRepair] SET NUMERIC_ROUNDABORT OFF
GO
ALTER DATABASE [EquipmentRepair] SET QUOTED_IDENTIFIER OFF
GO
ALTER DATABASE [EquipmentRepair] SET RECURSIVE_TRIGGERS OFF
GO
ALTER DATABASE [EquipmentRepair] SET  DISABLE_BROKER
GO
ALTER DATABASE [EquipmentRepair] SET AUTO_UPDATE_STATISTICS_ASYNC OFF
GO
ALTER DATABASE [EquipmentRepair] SET DATE_CORRELATION_OPTIMIZATION OFF
GO
ALTER DATABASE [EquipmentRepair] SET TRUSTWORTHY OFF
GO
ALTER DATABASE [EquipmentRepair] SET ALLOW_SNAPSHOT_ISOLATION OFF
GO
ALTER DATABASE [EquipmentRepair] SET PARAMETERIZATION SIMPLE
```



```

GO
ALTER DATABASE [EquipmentRepair] SET READ_COMMITTED_SNAPSHOT OFF
GO
ALTER DATABASE [EquipmentRepair] SET HONOR_BROKER_PRIORITY OFF
GO
ALTER DATABASE [EquipmentRepair] SET RECOVERY FULL
GO
ALTER DATABASE [EquipmentRepair] SET MULTI_USER
GO
ALTER DATABASE [EquipmentRepair] SET PAGE_VERIFY CHECKSUM
GO
ALTER DATABASE [EquipmentRepair] SET DB_CHAINING OFF
GO
ALTER DATABASE [EquipmentRepair] SET FILESTREAM( NON_TRANSACTED_ACCESS = OFF )
GO
ALTER DATABASE [EquipmentRepair] SET TARGET_RECOVERY_TIME = 60 SECONDS
GO
ALTER DATABASE [EquipmentRepair] SET DELAYED_DURABILITY = DISABLED
GO
ALTER DATABASE [EquipmentRepair] SET ACCELERATED_DATABASE_RECOVERY = OFF
GO
EXEC sys.sp_db_vardecimal_storage_format N'EquipmentRepair', N'ON'
GO
ALTER DATABASE [EquipmentRepair] SET QUERY_STORE = ON
GO
ALTER DATABASE [EquipmentRepair] SET QUERY_STORE (OPERATION_MODE = READ_WRITE,
CLEANUP_POLICY = (STALE_QUERY_THRESHOLD_DAYS = 30),
DATA_FLUSH_INTERVAL_SECONDS = 900, INTERVAL_LENGTH_MINUTES = 60,
MAX_STORAGE_SIZE_MB = 1000, QUERY_CAPTURE_MODE = AUTO, SIZE_BASED_CLEANUP_MODE
= AUTO, MAX_PLANS_PER_QUERY = 200, WAIT_STATS_CAPTURE_MODE = ON)
GO
USE [EquipmentRepair]
GO
/***** Object: Table [dbo].[Client]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Client](
    [client_id] [int] IDENTITY(1,1) NOT NULL,
    [client_name] [nvarchar](100) NULL,
    [client_surname] [nvarchar](100) NULL,
    [client_secondName] [nvarchar](100) NULL,
    [contact_info] [nvarchar](255) NULL,
PRIMARY KEY CLUSTERED
(
    [client_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Equipment]    Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Equipment](
    [equipment_id] [int] IDENTITY(1,1) NOT NULL,
    [serial_number] [nvarchar](50) NULL,
    [type] [nvarchar](100) NULL,
    [description] [nvarchar](255) NULL,

```

```

CONSTRAINT [PK__Equipmen__197068AF003BDF99] PRIMARY KEY CLUSTERED
(
    [equipment_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[FaultType]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[FaultType](
    [fault_type_id] [int] IDENTITY(1,1) NOT NULL,
    [description] [nvarchar](100) NULL,
PRIMARY KEY CLUSTERED
(
    [fault_type_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Position]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Position](
    [position_id] [int] IDENTITY(1,1) NOT NULL,
    [position_name] [nvarchar](100) NULL,
PRIMARY KEY CLUSTERED
(
    [position_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Priority]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Priority](
    [priority_id] [int] IDENTITY(1,1) NOT NULL,
    [priority_name] [nvarchar](50) NULL,
PRIMARY KEY CLUSTERED
(
    [priority_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Request]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON

```

```

GO
CREATE TABLE [dbo].[Request] (
    [request_id] [int] IDENTITY(1,1) NOT NULL,
    [date_created] [datetime] NULL,
    [date_ended] [datetime] NULL,
    [equipment_id] [int] NULL,
    [fault_type_id] [int] NULL,
    [problem_description] [nvarchar] (255) NULL,
    [client_id] [int] NULL,
    [technician_id] [int] NULL,
    [status_id] [int] NULL,
    [priority_id] [int] NULL,
    CONSTRAINT [PK__Request__18D3B90FF3D72648] PRIMARY KEY CLUSTERED
(
    [request_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[RequestLog]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[RequestLog] (
    [log_id] [int] IDENTITY(1,1) NOT NULL,
    [request_id] [int] NULL,
    [date_changed] [datetime] NULL,
    [comment] [nvarchar] (max) NULL,
    CONSTRAINT [PK__RequestL__9E2397E021172037] PRIMARY KEY CLUSTERED
(
    [log_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[Status]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Status] (
    [status_id] [int] IDENTITY(1,1) NOT NULL,
    [status_name] [nvarchar] (50) NULL,
    PRIMARY KEY CLUSTERED
(
    [status_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[Technician]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Technician] (
    [technician_id] [int] IDENTITY(1,1) NOT NULL,

```

```

        [technician_surname] [nvarchar](100) NULL,
        [technician_name] [nvarchar](100) NULL,
        [technician_secondName] [nvarchar](100) NULL,
        [position_id] [int] NULL,
        [contact_info] [varchar](255) NULL,
PRIMARY KEY CLUSTERED
(
        [technician_id] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object:  Table [dbo].[Users]      Script Date: 19.12.2024 16:06:09
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Users](
        [userID] [int] IDENTITY(1,1) NOT NULL,
        [Login] [nvarchar](100) NULL,
        [Password] [nvarchar](100) NULL,
        [TechnicianID] [int] NULL,
PRIMARY KEY CLUSTERED
(
        [userID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF, IGNORE_DUP_KEY = OFF,
ALLOW_ROW_LOCKS = ON, ALLOW_PAGE_LOCKS = ON, OPTIMIZE_FOR_SEQUENTIAL_KEY =
OFF) ON [PRIMARY]
) ON [PRIMARY]
GO
ALTER TABLE [dbo].[Request] ADD CONSTRAINT [DF__Request__date_cr__45F365D3]
DEFAULT (getdate()) FOR [date_created]
GO
ALTER TABLE [dbo].[RequestLog] ADD CONSTRAINT
[DF__RequestLo__date__4E88ABD4] DEFAULT (getdate()) FOR [date_changed]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__client__48CFD27E] FOREIGN KEY([client_id])
REFERENCES [dbo].[Client] ([client_id])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__client__48CFD27E]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__equipme__46E78A0C] FOREIGN KEY([equipment_id])
REFERENCES [dbo].[Equipment] ([equipment_id])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__equipme__46E78A0C]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__fault_t__47DBAE45] FOREIGN KEY([fault_type_id])
REFERENCES [dbo].[FaultType] ([fault_type_id])
ON DELETE SET NULL
GO
ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__fault_t__47DBAE45]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__priorit__4BAC3F29] FOREIGN KEY([priority_id])
REFERENCES [dbo].[Priority] ([priority_id])
ON DELETE SET NULL
GO

```

```

ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__priorit__4BAC3F29]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__status__4AB81AF0] FOREIGN KEY([status_id])
REFERENCES [dbo].[Status] ([status_id])
ON DELETE SET NULL
GO
ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__status__4AB81AF0]
GO
ALTER TABLE [dbo].[Request] WITH CHECK ADD CONSTRAINT
[FK__Request__technic__49C3F6B7] FOREIGN KEY([technician_id])
REFERENCES [dbo].[Technician] ([technician_id])
ON DELETE SET NULL
GO
ALTER TABLE [dbo].[Request] CHECK CONSTRAINT [FK__Request__technic__49C3F6B7]
GO
ALTER TABLE [dbo].[RequestLog] WITH CHECK ADD CONSTRAINT
[FK__RequestLo__reque__4F7CD00D] FOREIGN KEY([request_id])
REFERENCES [dbo].[Request] ([request_id])
ON DELETE CASCADE
GO
ALTER TABLE [dbo].[RequestLog] CHECK CONSTRAINT
[FK__RequestLo__reque__4F7CD00D]
GO
ALTER TABLE [dbo].[Technician] WITH CHECK ADD FOREIGN KEY([position_id])
REFERENCES [dbo].[Position] ([position_id])
ON DELETE SET NULL
GO
ALTER TABLE [dbo].[Users] WITH CHECK ADD FOREIGN KEY([TechnicianID])
REFERENCES [dbo].[Technician] ([technician_id])
GO
USE [master]
GO
ALTER DATABASE [EquipmentRepair] SET READ_WRITE
GO

```

Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»
(Финансовый университет)
Колледж информатики и программирования

ОТЧЕТ О ВЫПОЛНЕНИИ ЗАДАНИЯ № 2

**Тема: Разработка модулей программного обеспечения для компьютерных
систем**

Студент: Д.А. Мамонтов

Группа: 4ИСИП-421

Преподаватель: Альшакова Е.Л.

Дата: 18.12.2024

РАЗРАБОТКА МОДУЛЕЙ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ДЛЯ КОМПЬЮТЕРНЫХ СИСТЕМ

Цель работы: разработать WPF-приложение в соответствии с заданием, наполнить ее сущностями и атрибутами, написать листинг программ.

Было создано WPF-приложение для реализации графического интерфейса управления БД. Страница авторизации представлена на рисунке 12.

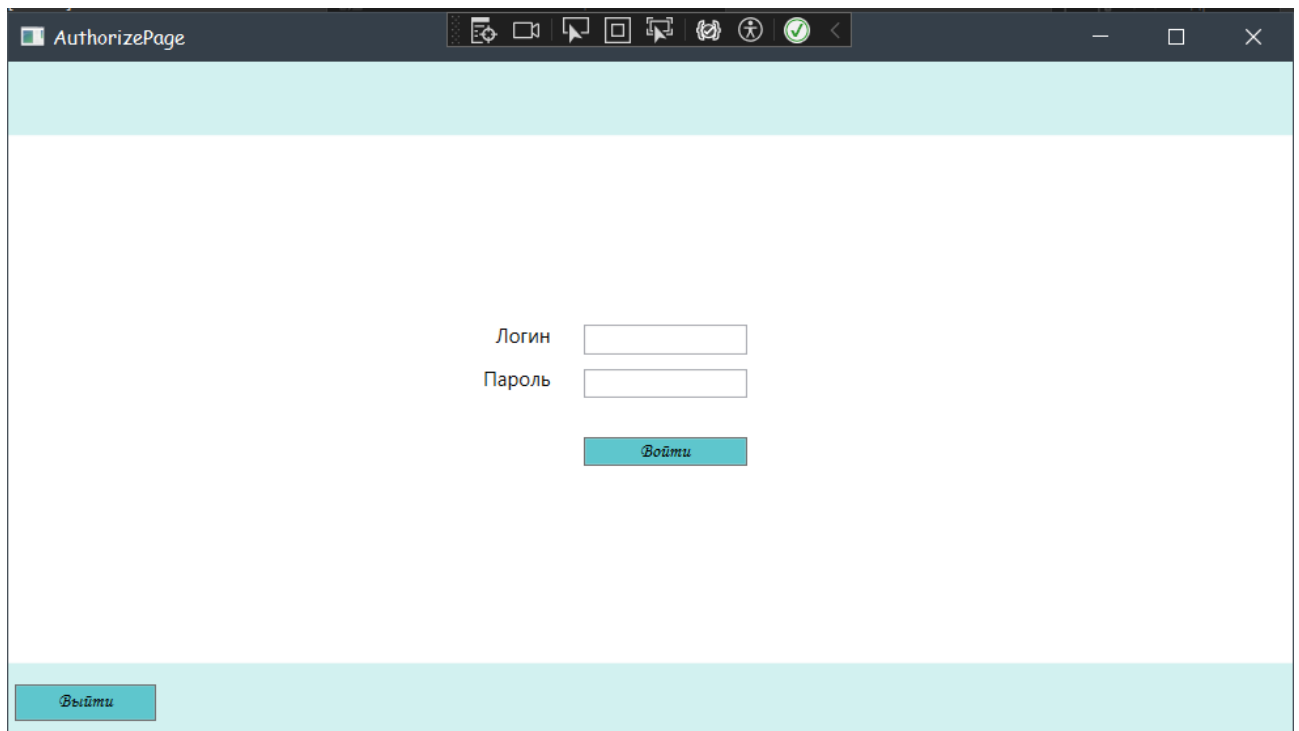


Рисунок 12 – Страница авторизации

Разметка окна “MainWindow” представлена в листинге 2.

Листинг 2 – Разметка окна “MainWindow”

```
<Window x:Class="Service.MainWindow"
        xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
        xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
        xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
        xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
        xmlns:local="clr-namespace:Service"
        mc:Ignorable="d"
        Title="MainWindow" Height="450" Width="800" Background="#D2F1F0">
    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="45" />
            <RowDefinition Height="*/>
            <RowDefinition Height="45"/>
        </Grid.RowDefinitions>
```

```

        <Frame x:Name="MainFrame" Grid.Row="1"
Source="Pages/AuthorizePage.xaml" Navigated="OnNavigated"
NavigationUIVisibility="Hidden" Background="White"/>
        <StackPanel x:Name="MenuContainer" Orientation="Horizontal"
Style="{StaticResource CustomStackPanel}" HorizontalAlignment="Center">
            <Button x:Name="GoBack" Content="Назад" Style="{StaticResource
CustomButton}" Width="125" Margin="30,0,0,0" Click="GoBack_Click" />
            <Button x:Name="Requests" Content="Заявки"
Style="{StaticResource CustomButton}" Width="125" Margin="30,0,0,0"
Click="Requests_Click"/>
            <Button x:Name="RequestsLog" Content="Логи изменений заявок"
Style="{StaticResource CustomButton}" Width="125" Margin="30,0,0,0"
Click="RequestsLog_Click"/>
            <Button x:Name="Reports" Content="Отчеты" Style="{StaticResource
CustomButton}" Width="125" Margin="30,0,0,0" Click="Reports_Click"/>
            <Button x:Name="QR" Content="QR-код" Style="{StaticResource
CustomButton}" Width="125" Margin="30,0,0,0" Click="QR_Click" />
        </StackPanel>
        <Button x:Name="Exit" Content="Выйти" Style="{StaticResource
CustomButton}" Margin="0,13,695,10" Click="Exit_Click" Grid.Row="2"
HorizontalAlignment="Right" Width="87"/>
    </Grid>
</Window>

```

Листинг 3 – Код для окна “MainWindow”

```

namespace Service
{
    public partial class MainWindow : Window
    {
        public MainWindow()
        {
            InitializeComponent();

            private void OnNavigated(object sender, NavigationEventArgs e)
            {
                if (!(e.Content is Page page)) return;
                this.Title = page.Title;

                if (page is Pages.AuthorizePage)
                    MenuContainer.Visibility = Visibility.Hidden;
                else MenuContainer.Visibility = Visibility.Visible;

                if (CurrentUser.positionID == 1)
                {
                    QR.Visibility = Visibility.Hidden;
                    RequestsLog.Visibility = Visibility.Visible;
                    Reports.Visibility = Visibility.Visible;

                }
                else
                {
                    QR.Visibility = Visibility.Visible;
                    RequestsLog.Visibility = Visibility.Hidden;
                    Reports.Visibility = Visibility.Hidden;

                }
            }

            private void Exit_Click(object sender, RoutedEventArgs e)
            {
                MainFrame.NavigationService.Navigate(new AuthorizePage());
            }
        }
    }
}

```



```

    }

    private void Requests_Click(object sender, RoutedEventArgs e)
    {
        MainFrame.NavigationService.Navigate(new Requests());
    }

    private void RequestsLog_Click(object sender, RoutedEventArgs e)
    {
        MainFrame.NavigationService.Navigate(new RequestLogs());
    }

    private void Reports_Click(object sender, RoutedEventArgs e)
    {
        MainFrame.NavigationService.Navigate(new ReportPage());
    }

    private void QR_Click(object sender, RoutedEventArgs e)
    {
        MainFrame.NavigationService.Navigate(new QRCodePage());
    }

    private void GoBack_Click(object sender, RoutedEventArgs e)
    {
        MainFrame.NavigationService.Navigate(new Requests());
    }
}
}

```

Разметка для страницы авторизации представлена в листинге 4.

Листинг 4 – Разметка страницы “AuthorizePage”

```

<Page x:Class="Service.Pages.AuthorizePage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:local="clr-namespace:Service.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="AuthorizePage">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="10*" />
            <RowDefinition Height="80*" />
            <RowDefinition Height="10*" />
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="353*" />
            <ColumnDefinition Width="447*" />
        </Grid.ColumnDefinitions>

        <StackPanel Grid.Row="1" Margin="0,0,10,0"
                    VerticalAlignment="Center" Height="104" HorizontalAlignment="Right"
                    Width="88">
            <Label Content="Логин" HorizontalAlignment="Right" />
            <Label Content="Пароль" HorizontalAlignment="Right" />
        </StackPanel>
    </Grid>

```

```

        <StackPanel Grid.Row="1" Grid.Column="1" HorizontalAlignment="Left"
Width="110" VerticalAlignment="Center" Height="91">
            <TextBox x:Name="Login" Width="100"/>
            <Separator Opacity="0" Height="5"/>
            <PasswordBox x:Name="Password" Width="100"/>
            <Separator Opacity="0" Height="20"/>
            <Button x:Name="Enter" Content="Войти" Width="100"
Click="Enter_Click" Style="{StaticResource CustomButton}"/>
        </StackPanel>
    </Grid>
</Page>

```

Листинг 5 – Код для страницы “AuthorizePage”

```

namespace Service.Pages
{
    public partial class AuthorizePage : Page
    {
        public AuthorizePage()
        {
            InitializeComponent();
        }

        private void Enter_Click(object sender, RoutedEventArgs e)
        {
            if (string.IsNullOrEmpty(Login.Text) ||
string.IsNullOrEmpty>Password.Password))
            {
                MessageBox.Show("Вы не заполнили все поля");
            }
            else
            {
                string login = Login.Text;
                string password = Password.Password;

                try
                {
                    using (var db = new Entities())
                    {
                        var user = db.Users
                            .AsNoTracking().FirstOrDefault(u => u.Login ==
login && u.Password == password);
                        if (user == null)
                        {
                            MessageBox.Show("Пользователь не
зарегистрирован");
                            return;
                        }
                        MessageBox.Show("Вы успешно авторизовались");

                        CurrentUser.positionID =
ReturnPositionID(user.TechnicianID);

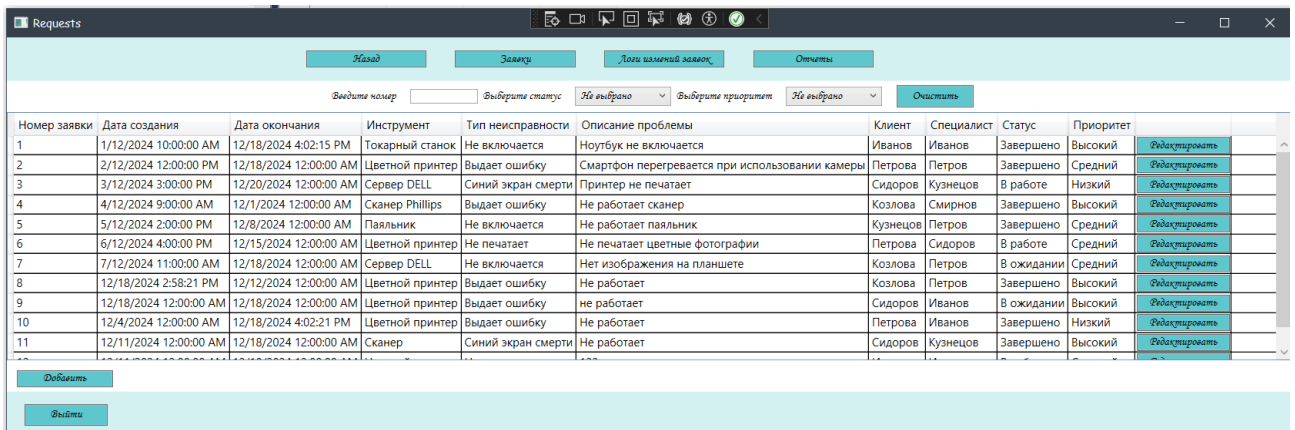
                        NavigationService.Navigate(new Requests());
                    }
                }
                catch (Exception ex)
                {
                    MessageBox.Show(ex.ToString());
                }
            }
        }
    }
}

```

```
private int ReturnPositionID(int? TechnicianID)
{
    var db = new Entities();
    var technician = db.Technician.AsNoTracking().FirstOrDefault(u
=> u.technician_id == TechnicianID);

    return (int)technician.position_id;
}
}
```

Отображение таблицы Request в приложении показан на рисунке 13.



Номер заявки	Дата создания	Дата окончания	Инструмент	Тип неисправности	Описание проблемы	Клиент	Специалист	Статус	Приоритет	
1	1/12/2024 10:00:00 AM	12/18/2024 4:02:15 PM	Токарный станок	Не включается	Ноутбук не включается	Иванов	Иванов	Завершено	Высокий	Редактировать
2	2/12/2024 12:00:00 PM	12/18/2024 12:00:00 AM	Цветной принтер	Выдает ошибку	Смартфон перегревается при использовании камеры	Петрова	Петров	Завершено	Средний	Редактировать
3	3/12/2024 3:00:00 PM	12/20/2024 12:00:00 AM	Сервер DELL	Синий экран смерти	Принтер не печатает	Сидоров	Кузнецов	В работе	Низкий	Редактировать
4	4/12/2024 9:00:00 AM	12/1/2024 12:00:00 AM	Сканер Phillips	Выдает ошибку	Не работает сканер	Козлова	Смирнов	Завершено	Высокий	Редактировать
5	5/12/2024 2:00:00 PM	12/8/2024 12:00:00 AM	Паяльник	Не включается	Не работает паяльник	Кузнецов	Петров	Завершено	Средний	Редактировать
6	6/12/2024 4:00:00 PM	12/15/2024 12:00:00 AM	Цветной принтер	Не печатает	Не печатает цветные фотографии	Петрова	Сидоров	В работе	Средний	Редактировать
7	7/12/2024 11:00:00 AM	12/18/2024 12:00:00 AM	Сервер DELL	Не включается	Нет изображения на планшете	Козлова	Петров	В ожидании	Средний	Редактировать
8	12/18/2024 2:58:21 PM	12/12/2024 12:00:00 AM	Цветной принтер	Выдает ошибку	Не работает	Козлова	Петров	Завершено	Высокий	Редактировать
9	12/18/2024 12:00:00 AM	12/18/2024 12:00:00 AM	Цветной принтер	Выдает ошибку	не работает	Сидоров	Иванов	В ожидании	Высокий	Редактировать
10	12/4/2024 12:00:00 AM	12/18/2024 4:02:21 PM	Цветной принтер	Выдает ошибку	Не работает	Петрова	Иванов	Завершено	Низкий	Редактировать
11	12/11/2024 12:00:00 AM	12/18/2024 12:00:00 AM	Сканер	Синий экран смерти	Не работает	Сидоров	Кузнецов	Завершено	Высокий	Редактировать

Рисунок 13 – Отображение таблицы Request

Отображение таблицы Request осуществляется на странице “Requests”. В приложении присутствует контроль роли пользователя. Например, пользователь с ролью менеджера сможет полностью редактировать все данные и проводить отчёты, а пользователи с другой ролью нет. Ниже представлены листинги для данной страницы.

Листинг 6 – Разметка страницы “Requests”

```
<Page x:Class="Service.Pages.Requests"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
xmlns:local="clr-namespace:Service.Pages"
mc:Ignorable="d"
d:DesignHeight="450" d:DesignWidth="800"
Title="Requests" IsVisibleChanged="Page_IsVisibleChanged_1">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="10*" />
            <RowDefinition Height="80*" />
            <RowDefinition Height="10*" />
        </Grid.RowDefinitions>
        <StackPanel Orientation="Horizontal" HorizontalAlignment="Center"
VerticalAlignment="Center">
```

```

        <Label Content="Введите номер" Style="{StaticResource AddLabel}"/>
        <TextBox Name="FindForNumber" Style="{StaticResource AddTextBox}"
Width="70" Margin="0,0,0,0" TextChanged="FindForNumber_TextChanged"/>
        <Label Content="Выберите статус" Style="{StaticResource AddLabel}"/>
        <ComboBox Name="FindForStatus" Style="{StaticResource AddComboBox}"
Width="100" Margin="0,0,0,0" SelectionChanged="FindForStatus_SelectionChanged"/>
        <Label Content="Выберите приоритет" Style="{StaticResource
AddLabel}"/>
        <ComboBox Name="FindForPriority" Style="{StaticResource
AddComboBox}" Width="100" Margin="0,0,15,0"
SelectionChanged="FindForPriority_SelectionChanged"/>
        <Button Name="ClearFilters" Content="Очистить"
Style="{StaticResource CustomButton}" Width="80" Click="ClearFilters_Click"/>
    </StackPanel>
    <DataGrid x:Name="DataGridRequests" AutoGenerateColumns="False"
IsReadOnly="True" Grid.Row="1">
        <DataGrid.Columns>
            <DataGridTextColumn Header="Номер заявки" Binding="{Binding
request_id}"/>
            <DataGridTextColumn Header="Дата создания" Binding="{Binding
date_created}"/>
            <DataGridTextColumn Header="Дата окончания" Binding="{Binding
date_ended}"/>
            <DataGridTextColumn Header="Инструмент" Binding="{Binding
Equipment.description}"/>
            <DataGridTextColumn Header="Тип неисправности" Binding="{Binding
FaultType.description}"/>
            <DataGridTextColumn Header="Описание проблемы" Binding="{Binding
problem_description}"/>
            <DataGridTextColumn Header="Клиент" Binding="{Binding
Client.client_surname}"/>
            <DataGridTextColumn Header="Специалист" Binding="{Binding
Technician.technician_surname}"/>
            <DataGridTextColumn Header="Статус" Binding="{Binding
Status.status_name}"/>
            <DataGridTextColumn Header="Приоритет" Binding="{Binding
Priority.priority_name}"/>
            <DataGridTemplateColumn Width="100">
                <DataGridTemplateColumn.CellTemplate>
                    <DataTemplate>
                        <Button Name="Edit" Content="Редактировать"
Style="{StaticResource CustomButton}" Click="Edit_Click"/>
                    </DataTemplate>
                </DataGridTemplateColumn.CellTemplate>
            </DataGridTemplateColumn>
        </DataGrid.Columns>
    </DataGrid>

    <Button x:Name="Add" Content="Добавить" Grid.Row="2"
HorizontalAlignment="Left" Margin="10, 10, 0, 0" VerticalAlignment="Top"
Style="{StaticResource CustomButton}" Width="100" Click="Add_Click"/>
</Grid>
</Page>

```

Листинг 7 – Код для страницы “Requests”

```

namespace Service.Pages
{
    public partial class Requests : Page
    {
        public Requests()
        {
            InitializeComponent();

            DataGridRequests.ItemsSource =
Entities.GetContext().Request.ToList();

```

```

        var priorityList = Entities.GetContext().Priority.Select(x =>
x.priority_name).ToList();
        priorityList.Insert(0, "Не выбрано");
        FindForPriority.ItemsSource = priorityList;

        var statusList = Entities.GetContext().Status.Select(x =>
x.status_name).ToList();
        statusList.Insert(0, "Не выбрано");
        FindForStatus.ItemsSource = statusList;
        FindForPriority.SelectedIndex = 0;
        FindForStatus.SelectedIndex = 0;

    }

    private void Add_Click(object sender, RoutedEventArgs e)
    {
        NavigationService.Navigate(new AddRequests(null, 0));
    }

    private void Edit_Click(object sender, RoutedEventArgs e)
    {
        NavigationService.Navigate(new AddRequests((sender as
Button).DataContext as Request, CurrentUser.positionID));
    }


    private void FindForNumber_TextChanged(object sender,
TextChangedEventArgs e)
    {
        UpdateRequests();
    }

    private void FindForStatus_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {
        UpdateRequests();
    }

    private void FindForPriority_SelectionChanged(object sender,
SelectionChangedEventArgs e)
    {
        UpdateRequests();
    }

    private void ClearFilters_Click(object sender, RoutedEventArgs e)
    {
        FindForNumber.Text = string.Empty;
        FindForStatus.SelectedIndex = 0;
        FindForPriority.SelectedIndex = 0;

        UpdateRequests();
    }

    private void UpdateRequests()
    {
        var currentRequests = Entities.GetContext().Request.ToList();

        currentRequests = currentRequests.Where(x =>
x.request_id.ToString().Contains(FindForNumber.Text)).ToList();

        if (FindForPriority.SelectedIndex != 0)
        {

```

```

        currentRequests = currentRequests.Where(x => x.priority_id
== FindForPriority.SelectedIndex).ToList();
    }

    if (FindForStatus.SelectedIndex != 0)
    {

        currentRequests = currentRequests.Where(x => x.status_id ==
FindForStatus.SelectedIndex).ToList();
    }
    DataGridRequests.ItemsSource = currentRequests;
}

private int ReturnPriority(string addPriority)
{
    return Entities.GetContext().Priority.Where(x => x.priority_name
== addPriority).Select(u => u.priority_id).FirstOrDefault();
}

private int ReturnStatus(string addStatus)
{
    return Entities.GetContext().Status.Where(x => x.status_name ==
addStatus).Select(u => u.status_id).FirstOrDefault();
}

private void Reports_Click(object sender, RoutedEventArgs e)
{
    NavigationService.Navigate(new ReportPage());
}

private void Page_IsVisibleChanged_1(object sender,
DependencyPropertyChangedEventArgs e)
{
    if (Visibility == Visibility.Visible)
    {

Entities.GetContext().ChangeTracker.Entries().ToList().ForEach(x =>
x.Reload());

        DataGridRequests.ItemsSource =
Entities.GetContext().Request.ToList();
    }
}
}
}

```

Добавление данных осуществляется на странице “AddRequests”. Демонстрация работы данной страницы представлена на рисунке 14.

Рисунок 14 – Демонстрация добавления данных

Листинг 8 – Разметка страницы “AddRequests”

```
<Page x:Class="Service.Pages.AddRequests"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:local="clr-namespace:Service.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="AddRequests">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
            <RowDefinition Height="*" />
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*" />
            <ColumnDefinition Width="*" />
        </Grid.ColumnDefinitions>

        <Label Content="Номер заявки" Style="{StaticResource AddLabel}" />
        <TextBox x:Name="NumberRequest" Text="{Binding request_id}"
            Grid.Column="1" Style="{StaticResource AddTextBox}" Width="200" />
    </Grid>
</Page>
```

```

        <Label Content="Дата создания заявки" Grid.Row="1"
Style="{StaticResource AddLabel}"/>

        <DatePicker Name="DateCreate" SelectedDate="{Binding date_created}"
SelectedDateFormat="Short" Grid.Row="1" Grid.Column="1"
VerticalAlignment="Center" Width="200" HorizontalAlignment="Left"
Margin="10,0,0,0"/>

        <Label Content="Дата выполнения заявки" Grid.Row="2"
Style="{StaticResource AddLabel}"/>
        <DatePicker Name="DateEnd" SelectedDate="{Binding date_ended}"
IsDropDownOpen="False" SelectedDateFormat="Short" Grid.Row="2"
Grid.Column="1" VerticalAlignment="Center" Width="200"
HorizontalAlignment="Left" Margin="10,0,0,0"/>

        <Label Content="Инструмент" Grid.Row="3" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="equipment" SelectedItem="{Binding
Equipment.description}" Grid.Row="3" Grid.Column="1" Width="200"
Style="{StaticResource AddComboBox}" IsEnabled="True" />

        <Label Content="Тип ошибки" Grid.Row="4" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="faultType" Text="{Binding FaultType.description}"
Grid.Column="1" Grid.Row="4" Width="200" Style="{StaticResource
AddComboBox}" IsEnabled="True" IsEditable="True"/>

        <Label Content="Описание проблемы" Grid.Row="5"
Style="{StaticResource AddLabel}"/>
        <TextBox x:Name="descriptionProblem" Text="{Binding
problem_description}" Grid.Column="1" Grid.Row="5" Style="{StaticResource
AddTextBox}" Width="200" IsEnabled="True" />

        <Label Content="Клиент" Grid.Row="6" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="client" SelectedItem="{Binding
Client.client_surname}" Grid.Column="1" Grid.Row="6" Style="{StaticResource
AddComboBox}" Width="200" />

        <Label Content="Специалист" Grid.Row="7" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="technician" SelectedItem="{Binding
Technician.technician_surname}" Grid.Column="1" Grid.Row="7"
Style="{StaticResource AddComboBox}" Width="200" />

        <Label Content="Статус" Grid.Row="8" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="status" SelectedItem="{Binding
Status.status_name}" Grid.Row="8" Grid.Column="1" Style="{StaticResource
AddComboBox}" Width="200" />

        <Label Content="Приоритет" Grid.Row="9" Style="{StaticResource
AddLabel}"/>
        <ComboBox x:Name="priority" SelectedItem="{Binding
Priority.priority_name}" Grid.Column="1" Grid.Row="9" Style="{StaticResource
AddComboBox}" Width="200" />

        <Button x:Name="Cansel" Content="Отмена" Style="{StaticResource
CustomButton}" Grid.Row="10" HorizontalAlignment="Right"
VerticalAlignment="Center" Width="200" Margin="0,0,10,0"
Click="Cansel_Click"/>
        <Button x:Name="Add" Content="Добавить" Style="{StaticResource
CustomButton}" Grid.Row="10" Grid.Column="1" HorizontalAlignment="Left"
VerticalAlignment="Center" Width="200" Margin="10,0,0,0" Click="Add_Click"/>

```



```
</Grid>
</Page>
```

Листинг 9 – Код для страницы “AddRequests”

```
namespace Service.Pages
{
    public partial class AddRequests : Page
    {
        private Request requests = new Request();

        public AddRequests(Request selectRequest, int roleID)
        {
            InitializeComponent();

            if (selectRequest != null)
            {
                Add.Content = "Сохранить";

                requests = selectRequest;

                if (roleID == 1)
                {
                }
                else
                {
                    status.IsEnabled = false;
                    equipment.IsEnabled = false;
                    faultType.IsEnabled = false;
                    client.IsEnabled = false;
                    priority.IsEnabled = false;
                    DateCreate.IsEnabled = false;
                    NumberRequest.IsReadOnly = true;
                }
            }

            equipment.ItemsSource = Entities.GetContext().Equipment.Select(u
=> u.description).ToList();
            faultType.ItemsSource = Entities.GetContext().FaultType.Select(u
=> u.description).ToList();
            client.ItemsSource = Entities.GetContext().Client.Select(u =>
u.client_surname).ToList();
            technician.ItemsSource =
Entities.GetContext().Technician.Select(u => u.technician_surname).ToList();
            status.ItemsSource = Entities.GetContext().Status.Select(u =>
u.status_name).ToList();
            priority.ItemsSource = Entities.GetContext().Priority.Select(u
=> u.priority_name).ToList();

            DataContext = requests;
        }

        private void Cansel_Click(object sender, RoutedEventArgs e)
        {
            NavigationService.Navigate(new Requests());
        }

        private void Add_Click(object sender, RoutedEventArgs e)
        {

```

```

        string dateTime = DateCreate.Text;
        string endTime = DateEnd.Text;
        string addNumberRequest = NumberRequest.Text;
        string addEquipment = equipment.Text;
        string addFaultType = faultType.Text;
        string addStatus = status.Text;
        string addPriority = priority.Text;
        string addProblemDescription = descriptionProblem.Text;
        string addClient = client.Text;
        string addTechnician = technician.Text;

        if (string.IsNullOrEmpty(dateTime) ||
            string.IsNullOrEmpty(addClient) || string.IsNullOrEmpty(addStatus) ||
            string.IsNullOrEmpty(addPriority) ||
                string.IsNullOrEmpty(addProblemDescription) ||
            string.IsNullOrEmpty(addTechnician) ||
            string.IsNullOrEmpty(addNumberRequest) ||
                string.IsNullOrEmpty(addEquipment) ||
            string.IsNullOrEmpty(addFaultType) || string.IsNullOrEmpty(endTime))
        { MessageBox.Show("Вы не заполнили все поля! Попробуйте ещё раз"); }
        else
        {
            if (DateTime.TryParse(endTime, out DateTime dateEnd) &&
                (DateTime.TryParse(dateTime, out DateTime dateCreate)) &&
                int.TryParse(addNumberRequest, out int numberRequest))
            {
                requests.date_created = dateCreate;
                requests.date_ended = dateEnd;
                requests.priority_id = ReturnPriority(addPriority);
                requests.status_id = ReturnStatus(addStatus);
                requests.equipment_id = ReturnEquipment(addEquipment);
                requests.problem_description = addProblemDescription;
                requests.request_id = int.Parse(addNumberRequest);
                requests.fault_type_id = ReturnFaultType(addFaultType);
                requests.client_id = ReturnClient(addClient);
                requests.technician_id = ReturnTehnician(addTechnician);

                var request = Entities.GetContext().Request.Where(x =>
                    x.request_id == requests.request_id).FirstOrDefault();
                if (request == null)
                {
                    Entities.GetContext().Request.Add(requests);
                    try
                    {
                        Entities.GetContext().SaveChanges();
                        MessageBox.Show("Заявка добавлена");
                        NavigationService.Navigate(new Requests());
                    }
                    catch (Exception ex)
                    {
                        MessageBox.Show(ex.Message.ToString());
                    }
                }
                else
                {
                    try
                    {
                        var db = new Entities();
                        var currentRequest = db.Request.Where(x =>
                            x.request_id == requests.request_id).FirstOrDefault();

                        if (currentRequest != null)
                        {

```

```

requests.technician_id;
requests.client_id;
requests.problem_description;
requests.equipment_id;
requests.fault_type_id;
requests.status_id;

ReturnStatus("Завершено"))

DateTime.Now;

requests.date_ended;

currentRequest.technician_id =
currentRequest.client_id =
currentRequest.problem_description =
currentRequest.equipment_id =
currentRequest.fault_type_id =
currentRequest.status_id =

if (requests.status_id ==
{
    currentRequest.date_ended =
}
else
{
    currentRequest.date_ended =
}

RequestLog requestLog = new RequestLog()
{
    request_id = requests.request_id,
    date_changed = DateTime.Now,
};

db.RequestLog.Add(requestLog);
db.SaveChanges(); // Важно сохранить

изменения
данные");

MessageBox.Show("Вы успешно изменили

NavigationService.Navigate(new Requests());
}
else
{
    MessageBox.Show("Запись не найдена.");
}
}
catch (Exception ex)
{
    MessageBox.Show($"Произошла ошибка:
{ex.Message}");
}
}

}

private int ReturnPriority(string addPriority)
{
    return Entities.GetContext().Priority.Where(x => x.priority_name
== addPriority).Select(u => u.priority_id).FirstOrDefault();
}

private int ReturnStatus(string addStatus)
{
    return Entities.GetContext().Status.Where(x => x.status_name ==
addStatus).Select(u => u.status_id).FirstOrDefault();
}

```

```

        private int ReturnEquipment(string addEquipment)
        {
            return Entities.GetContext().Equipment.Where(x => x.description
== addEquipment).Select(u => u.equipment_id).FirstOrDefault();
        }

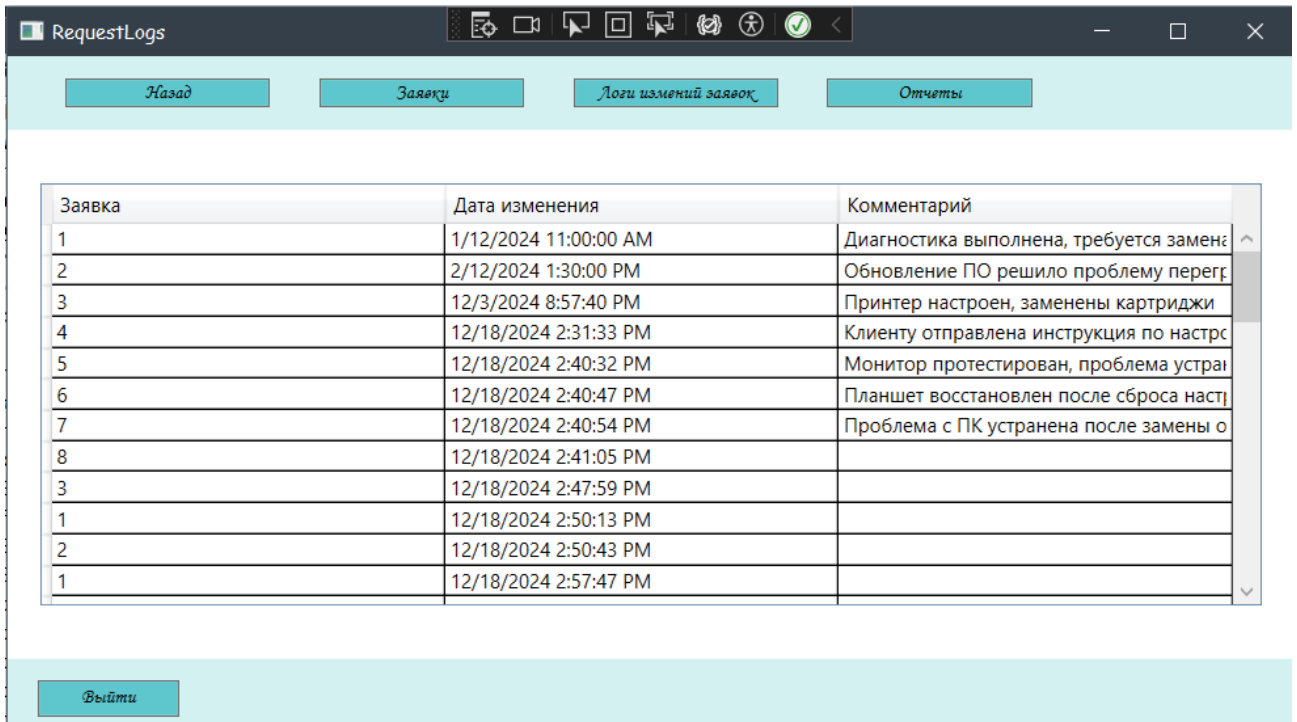
        private int ReturnFaultType(string addFaultType)
        {
            return Entities.GetContext().FaultType.Where(x => x.description
== addFaultType).Select(u => u.fault_type_id).FirstOrDefault();
        }

        private int ReturnClient(string addClient)
        {
            return Entities.GetContext().Client.Where(x => x.client_surname
== addClient).Select(u => u.client_id).FirstOrDefault();
        }

        private int ReturnTehnician(string addTehnician)
        {
            return Entities.GetContext().Technician.Where(x =>
x.technician_surname == addTehnician).Select(u =>
u.technician_id).FirstOrDefault();
        }
    }
}

```

В программе реализованы логи изменений заявок. Это представлено на рисунке 15 и реализовано на странице “RequestLogs”.



The screenshot shows a web application window titled "RequestLogs". It has a navigation bar with four buttons: "Назад", "Заявки", "Логи изменений заявок", and "Отчеты". Below the navigation bar is a table with three columns: "Заявка", "Дата изменения", and "Комментарий". The table contains 14 rows of data. At the bottom of the window, there is a "Выйти" button.

Заявка	Дата изменения	Комментарий
1	1/12/2024 11:00:00 AM	Диагностика выполнена, требуется замена
2	2/12/2024 1:30:00 PM	Обновление ПО решило проблему перегрева
3	12/3/2024 8:57:40 PM	Принтер настроен, заменены картриджи
4	12/18/2024 2:31:33 PM	Клиенту отправлена инструкция по настройке
5	12/18/2024 2:40:32 PM	Монитор протестирован, проблема устранена
6	12/18/2024 2:40:47 PM	Планшет восстановлен после сброса настроек
7	12/18/2024 2:40:54 PM	Проблема с ПК устранена после замены оперативной памяти
8	12/18/2024 2:41:05 PM	
3	12/18/2024 2:47:59 PM	
1	12/18/2024 2:50:13 PM	
2	12/18/2024 2:50:43 PM	
1	12/18/2024 2:57:47 PM	

Рисунок 15 – Демонстрация страницы с логами изменений
заявок

Листинги для данной страницы представлены ниже:

Листинг 10 – Разметка страницы “RequestLogs”

```
<Page x:Class="Service.Pages.RequestLogs"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:local="clr-namespace:Service.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="RequestLogs">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="10*" />
            <RowDefinition Height="80*" />
            <RowDefinition Height="10*" />
        </Grid.RowDefinitions>
        <DataGrid x:Name="DGRequestLogs" AutoGenerateColumns="False"
HorizontalAlignment="Center" Grid.Row="1" Margin="20,0,20,0">
            <DataGrid.Columns>
                <DataGridTextColumn Header="Заявка" Binding="{Binding
Request.request_id}" Width="*" />
                <DataGridTextColumn Header="Дата изменения"
Binding="{Binding date_changed}" Width="*" />
                <DataGridTextColumn Header="Комментарий" Binding="{Binding
comment}" Width="*" />
            </DataGrid.Columns>
        </DataGrid>

    </Grid>
</Page>
```

Листинг 11 – Код для страницы “RequestLogs”

```
namespace Service.Pages
{
    public partial class RequestLogs : Page
    {
        public RequestLogs()
        {
            InitializeComponent();

            DGRequestLogs.ItemsSource =
Entities.GetContext().RequestLog.ToList();
        }
    }
}
```

В приложении реализованы отчёты на странице “ReportPage”. Пример подсчёта количества выполненных заявок представлен на рисунке 16.

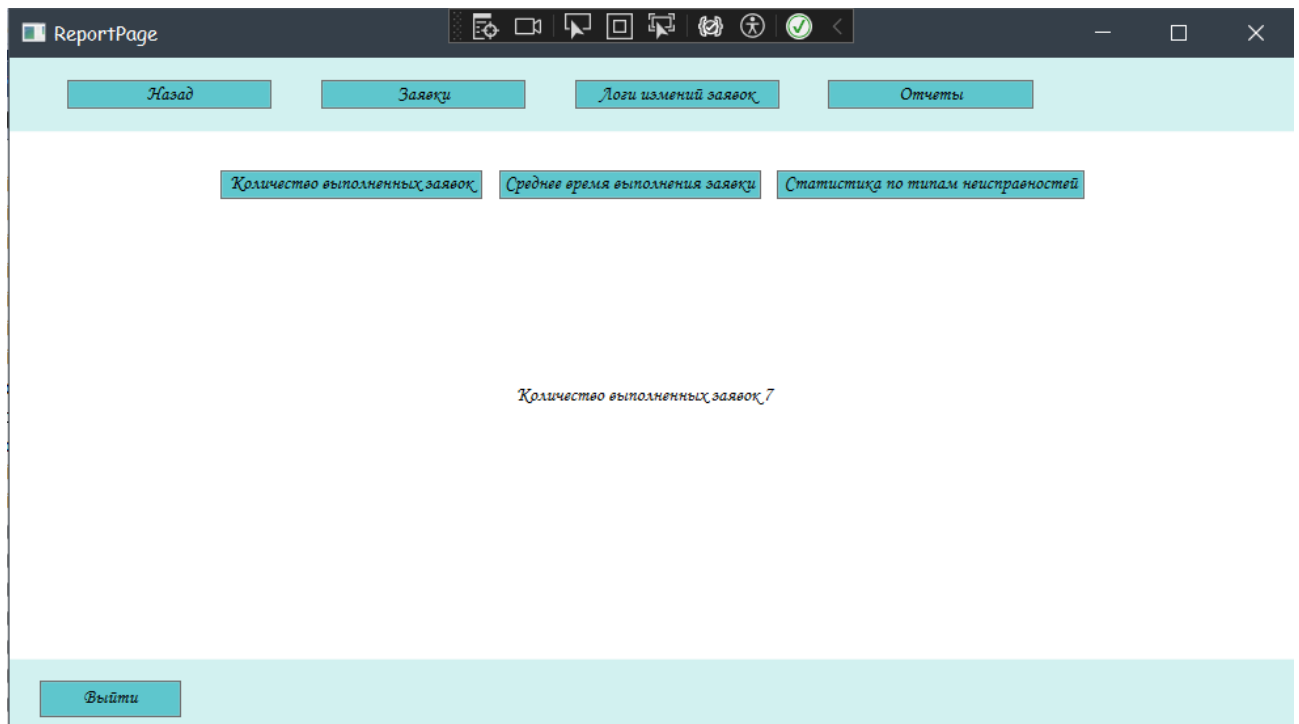


Рисунок 16 – Демонстрация страницы с отчётами

Листинги для данной страницы представлены ниже:

Листинг 12 – Разметка страницы “ReportPage”

```
<Page x:Class="Service.Pages.ReportPage"
      xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
      xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
      xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
      xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
      xmlns:local="clr-namespace:Service.Pages"
      mc:Ignorable="d"
      d:DesignHeight="450" d:DesignWidth="800"
      Title="ReportPage">

    <Grid>
        <Grid.RowDefinitions>
            <RowDefinition Height="20*" />
            <RowDefinition Height="60*" />
            <RowDefinition Height="20*" />
        </Grid.RowDefinitions>
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*" />
            <ColumnDefinition Width="*" />
        </Grid.ColumnDefinitions>

        <StackPanel Style="{StaticResource CustomStackPanel}"
            Orientation="Horizontal" Grid.ColumnSpan="2">
            <Button x:Name="QuantityCompletedRequestsButton"
                Content="Количество выполненных заявок" Style="{StaticResource CustomButton}" Click="QuantityCompletedRequestsButton_Click" Width="160"/>
            <Separator Opacity="0" Width="10"/>
            <Button x:Name="AVGTimeCompletedRequestsButton" Content="Среднее время выполнения заявки" Style="{StaticResource CustomButton}" Click="AVGTimeCompletedRequestsButton_Click" Width="160"/>
            <Separator Opacity="0" Width="10"/>
        </StackPanel>
    </Grid>
</Page>
```

```

        <Button x:Name="StatisticsOnTypesOfFaultsButton"
Content="Статистика по типам неисправностей" Style="{StaticResource
CustomButton}" Click="StatisticsOnTypesOfFaultsButton_Click" Width="188"/>
    </StackPanel>
    <Label x:Name="QuantityCompletedRequestsLabel"
Style="{StaticResource AddLabel}" Grid.Row="1" Grid.ColumnSpan="2"
HorizontalAlignment="Center"/>
    <Label x:Name="AVGTimeCompletedRequestsLabel" Style="{StaticResource
AddLabel}" Grid.Row="1" Grid.ColumnSpan="2" HorizontalAlignment="Center"/>

    <StackPanel x:Name="StatisticsLabelsContainer"
Style="{StaticResource CustomStackPanel}" Grid.Row="1" Grid.ColumnSpan="2"/>
</Grid>

</Page>

```

Листинг 13 – Код для страницы “ReportPage”

```

namespace Service.Pages
{
    public partial class ReportPage : Page
    {
        string defaultQCPCContent;
        int FaultTypeCount;
        int RequestCount;
        public ReportPage()
        {
            InitializeComponent();

            defaultQCPCContent =
QuantityCompletedRequestsButton.Content.ToString();

        }
        private void AVGTimeCompletedRequestsButton_Click(object sender,
RoutedEventArgs e)
        {
            ClearPage();
            AVGTimeCompletedRequestsLabel.Visibility = Visibility.Visible;

            RequestCount = Entities.GetContext().Request.Count();
            float counter = 0;
            float time = 0;

            for (int i = 1; i <= RequestCount; i++)
            {
                DateTime startDate =
(DateTime)Entities.GetContext().Request.Where(x => x.request_id ==
i).Select(u => u.date_created).FirstOrDefault();
                var middleDate = Entities.GetContext().Request.Where(x =>
x.request_id == i && x.status_id == 3).Select(u =>
u.date_ended).FirstOrDefault();
                DateTime endDate;
                if (middleDate == null)
                {
                    endDate = DateTime.Now;
                }
                else { endDate = (DateTime)middleDate; }
                time += (endDate - startDate).Days;
                counter++;
            }
            AVGTimeCompletedRequestsLabel.Content = "Среднее время
выполнение заявки " + (time / counter).ToString() + " дней";
        }
    }
}

```

```

        private void QuantityCompletedRequestsButton_Click(object sender,
RoutedEventArgs e)
        {
            ClearPage();
            QuantityCompletedRequestsLabel.Visibility = Visibility.Visible;
            QuantityCompletedRequestsLabel.Content = defaultQCPContent + " "
+ (Entities.GetContext().Request.Where(x => x.status_id ==
3).Count().ToString());
        }

        private void StatisticsOnTypesOfFaultsButton_Click(object sender,
RoutedEventArgs e)
        {
            ClearPage();

            FaultTypeCount = Entities.GetContext().FaultType.Count();

            for (int i = 1; i <= FaultTypeCount; i++)
            {
                FillingLabelsContent(i.ToString());
            }

            private void FillingLabelsContent(string labelName)
            {
                int id = int.Parse(labelName);
                string FaultTypeName = Entities.GetContext().FaultType.Where(x
=> x.fault_type_id == id).Select(u => u.description).FirstOrDefault();
                string FaultTypeCount = Entities.GetContext().Request.Where(x =>
x.fault_type_id == id).Count().ToString();
                Label label = new Label()
                {
                    Name = "FaultTypeQuantity" + labelName,
                    Content = FaultTypeName + " " + FaultTypeCount,
                    FontSize = 12,
                    FontFamily = new System.Windows.Media.FontFamily("Monotype
Corsiva"),
                    HorizontalAlignment = HorizontalAlignment.Center,
                    VerticalAlignment = VerticalAlignment.Center,
                };
                StatisticsLabelsContainer.Children.Add(label);
            }

            private void ClearPage()
            {
                StatisticsLabelsContainer.Children.Clear();
                QuantityCompletedRequestsLabel.Visibility = Visibility.Hidden;
                AVGTimeCompletedRequestsLabel.Visibility = Visibility.Hidden;
            }
        }
    }
}

```

В приложении так же реализована страница с QR-кодом. Демонстрация работы данной страницы “QRCodePage” представлена ниже на рисунке 17.



Рисунок 17 – Демонстрация страницы с QR-кодом

Листинги для данной страницы представлены ниже:

Листинг 14 – Разметка страницы “QRCodePage”

```
<Page x:Class="Service.Pages.QRCodePage"
  xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
  xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml"
  xmlns:mc="http://schemas.openxmlformats.org/markup-compatibility/2006"
  xmlns:d="http://schemas.microsoft.com/expression/blend/2008"
  xmlns:local="clr-namespace:Service.Pages"
  mc:Ignorable="d"
  d:DesignHeight="450" d:DesignWidth="800"
  Title="QRCodePage">

  <Grid>
    <Grid.ColumnDefinitions>
      <ColumnDefinition Width="200"/>
      <ColumnDefinition Width="*/>
      <ColumnDefinition Width="200"/>
    </Grid.ColumnDefinitions>
    <Grid.RowDefinitions>
      <RowDefinition Height="57"/>
      <RowDefinition Height="286"/>
      <RowDefinition/>
    </Grid.RowDefinitions>
    <StackPanel Grid.Row="1" Grid.Column="1" Orientation="Vertical">
      <Image Name="imgbarcode"/>
      <TextBlock Name="tbkbarcodecontent" FontWeight="Bold" HorizontalAlignment="Center"/>
    </StackPanel>
  </Grid>
</Page>
```

Листинг 15 – Код для страницы “QRCodePage”

```
namespace Service.Pages
{
    public partial class QRCodePage : Page
    {
        public QRCodePage()
        {
            InitializeComponent();
        }
    }
}
```

```

        {
            InitializeComponent();

GenerateQR("https://forms.yandex.ru/u/6762b385f47e735ec9193d0a/");
        }

        private void GenerateQR(string content)
        {
            try
            {
                System.Drawing.Image img = null;
                BitmapImage bimg = new BitmapImage();
                using (var ms = new MemoryStream())
                {
                    BarcodeWriter writer;
                    writer = new ZXing.BarcodeWriter() { Format =
BarcodeFormat.QR_CODE };
                    writer.Options.Height = 120;
                    writer.Options.Width = 200;
                    writer.Options.PureBarcode = true;
                    img = writer.Write(content);
                    img.Save(ms, System.Drawing.Imaging.ImageFormat.Jpeg);
                    ms.Position = 0;
                    bimg.BeginInit();
                    bimg.CreateOptions =
BitmapCreateOptions.PreservePixelFormat;
                    bimg.CacheOption = BitmapCacheOption.OnLoad;
                    bimg.UriSource = null;
                    bimg.StreamSource = ms;
                    bimg.EndInit();
                    this.imgbarcode.Source = bimg;
                    this.tbkbarcodecontent.Text = content;
                }
            }
            catch (Exception ex)
            {
                MessageBox.Show(ex.Message);
            }
        }
    }
}

```

Для стилизации приложения был создан словарь стилей. Код для него представлен ниже в листинге 16.

Листинг 15 – словарь стилей “QRCodePage”

```

<ResourceDictionary
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation"
xmlns:x="http://schemas.microsoft.com/winfx/2006/xaml">
    <Style x:Key="CustomPasswordBox"
        TargetType="PasswordBox">
        <Setter Property="Margin" Value="10,0,0,0"/>
        <Setter Property="VerticalAlignment" Value="Center"/>
        <Setter Property="HorizontalAlignment" Value="Left"/>
        <Setter Property="FontSize" Value="12"/>
    </Style>

    <Style x:Key="CustomButton"

```

```

        TargetType="Button">
        <Setter Property="FontSize" Value="12"/>
        <Setter Property="Background" Value="#5EC6CD"/>
        <Setter Property="Foreground" Value="Black"/>
        <Setter Property="FontFamily" Value="Monotype Corsiva"/>
    </Style>
    <Style x:Key="CustomStackPanel"
        TargetType="StackPanel">
        <Setter Property="HorizontalAlignment" Value="Center"/>
        <Setter Property="VerticalAlignment" Value="Center"/>

    </Style>
    <Style x:Key="AddLabel"
        TargetType="Label">
        <Setter Property="Margin" Value="0,0,10,0"/>
        <Setter Property="VerticalAlignment" Value="Center"/>
        <Setter Property="HorizontalAlignment" Value="Right"/>
        <Setter Property="FontSize" Value="12"/>
        <Setter Property="FontFamily" Value="Monotype Corsiva"/>
    </Style>

    <Style x:Key="AddComboBox"
        TargetType="ComboBox">
        <Setter Property="VerticalAlignment" Value="Center"/>
        <Setter Property="HorizontalAlignment" Value="Left"/>
        <Setter Property="Margin" Value="10,0,0,0"/>
        <Setter Property="FontFamily" Value="Monotype Corsiva"/>
    </Style>

    <Style x:Key="AddTextBox"
        TargetType="TextBox">
        <Setter Property="Margin" Value="10,0,0,0"/>
        <Setter Property="VerticalAlignment" Value="Center"/>
        <Setter Property="HorizontalAlignment" Value="Left"/>
        <Setter Property="FontSize" Value="12"/>
        <Setter Property="FontFamily" Value="Monotype Corsiva"/>
    </Style>

    <Style x:Key="ReportTextBlock"
        TargetType="TextBlock">
        <Setter Property="FontSize" Value="12"/>

    </Style>

```

```

</ResourceDictionary>

```